

## 13.56MHz 非接触读卡器 IC

### 1 产品概述

MFRC522 是一款应用广泛的非接触读卡器芯片，集成了在 13.56MHz 下的多种非接触通信方式和协议，具有很高的技术集成度。

### 2 功能特性

- ◆ 读写器模式支持 ISO/IEC 14443A 标准
- ◆ 高度集成的 解调和解码模拟电路
- ◆ 只需很少的外部器件，即可将输出驱动连接至天线
- ◆ 在读写器模式下，典型操作距离最高可达 70mm，具体取决于天线设计和电源
- ◆ 支持的主机接口
  - I2C 接口，快速模式速率可达 400k，高速模式可达 3400k
  - UART 接口，最高可达 1228.8k
  - SPI 接口，速率最高 10Mbit/s
- ◆ 64 字节 FIFO 缓冲器用于接收和发送
- ◆ 灵活的中断模式
- ◆ 低功耗的硬件掉电模式
- ◆ 支持软件掉电模式
- ◆ 支持 LPCD 功能
- ◆ 可编程定时器
- ◆ 内部振荡器，连接 27.12MHz 晶体
- ◆ 2.5V-5V 宽范围电源电压
- ◆ CRC 协处理器
- ◆ 可编程 I/O 管脚
- ◆ 内部自检功能

### 3 应用领域

应用领域广泛，在考勤签到、门禁控制、公共交通、食堂就餐、水电气充值、便携式手持设备、各种会员系统等多方面的综合应用，有很强的系统应用扩展性。

## 4 引脚配置和功能

图 1 QFN32 封装管脚图

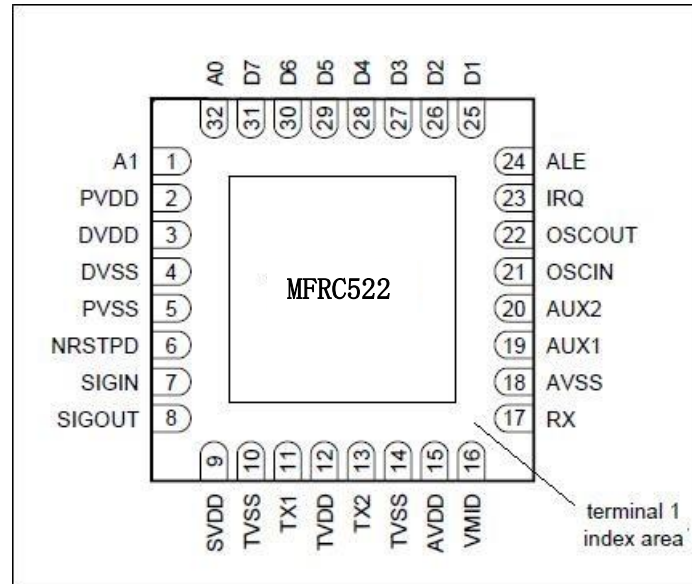


表 1 管脚说明

管脚号	管脚名	类型	管脚描述
1	A1	I	地址线
2	PVDD	P	管脚电源
3	DVDD	P	数字电源
4	DVSS	G	数字地
5	PVSS	G	管脚电源地
6	NRSTPD	I	复位脚：为低电平时，内部功能模块包括振荡器均停止工作，输入管脚与外部断开。 该管脚上的上升沿可用来开启内部复位相位。
7	SIGIN	I	通信接口输入：接收数字数据流、串行数据流
8	SIGOUT	O	通信接口输出：输出串行数据流
9	SVDD	P	S <sup>2</sup> C 管脚电源：向 S <sup>2</sup> C 管脚供电
10	TVSS	G	发送器 TX1 和 TX2 输出级的地
11	TX1	O	载波发送管脚 1
12	TVDD	P	发送驱动器电源
13	TX2	O	载波发送管脚 2
14	TVSS	G	发送驱动器电源地

管脚号	管脚名	类型	管脚描述
15	AVDD	P	模拟电源
16	VMID	P	内部参考电压
17	RX	I	RF 信号输入
18	AVSS	G	模拟地
19	AUX1	O	用于测试的辅助输出
20	AUX2	O	用于测试的辅助输出
21	OSCIN	I	晶振反相放大器输入；也是外部时钟的输入。 该管脚还可用作外部时钟（fosc=27.12MHz）的输入
22	OSCON	O	晶振反相放大器输出
23	IRQ	O	中断请求输出：指示一个中断事件
24	ALE	I	地址锁存使能：高电平时将 ADO-AD5 锁存到内部地址锁存
25-31	D1-D7	I/O	8 位双向数据总线 注：不支持 8 位并行接口 注：如果主控制器选择 I <sup>2</sup> C 作为数字主控制器接口，那么这些管脚可以用来定义 PC 地址 注：对于串行接口，这些管脚可以用作测试信号或 I/O
32	A0	I	地址线

## 5 功能框图

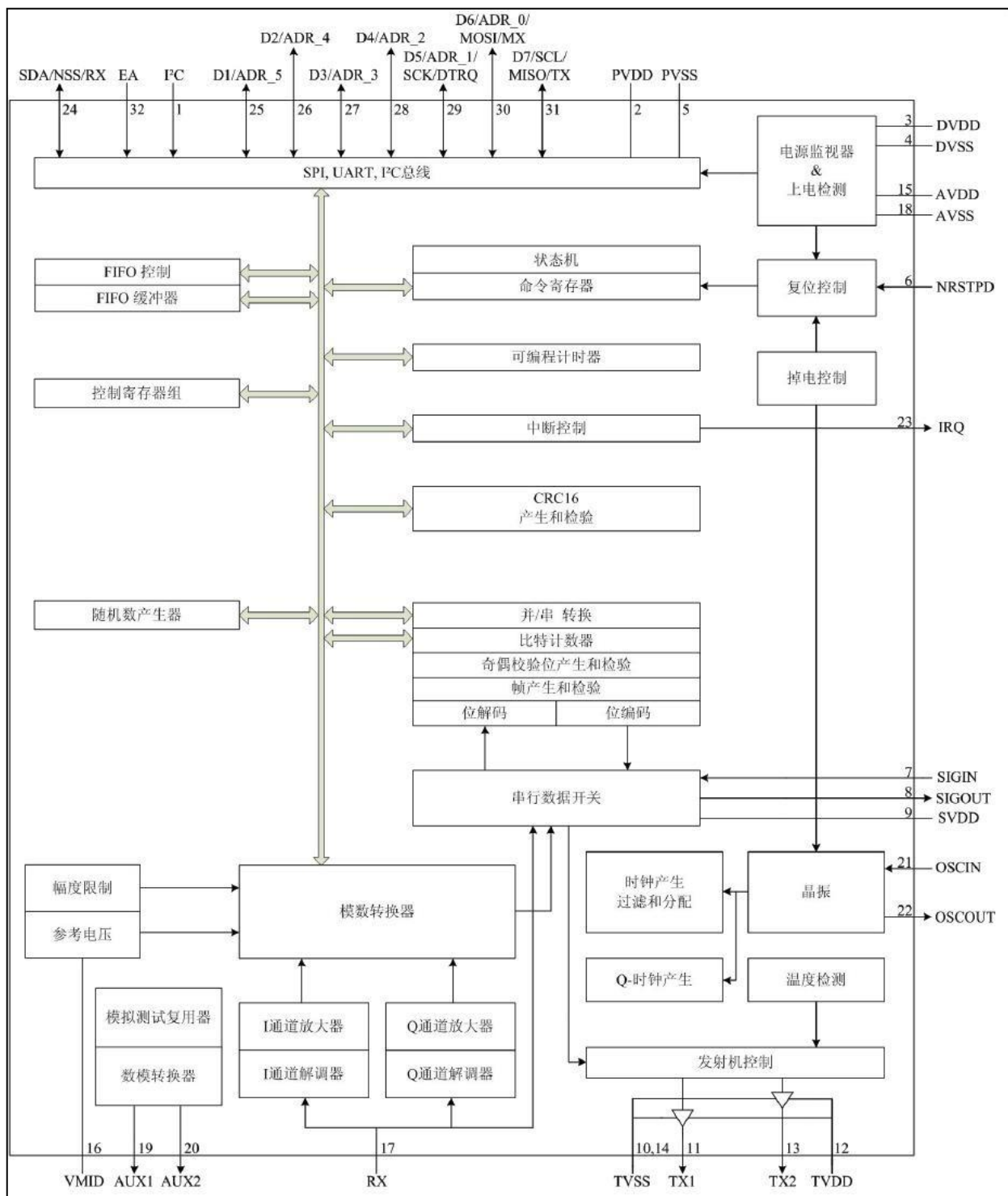


图 2 功能框图

## 6 功能描述

MFRC522 工作在读写器模式时，其传输模块支持 ISO/IEC 14443 A 标准，并可采用多种传输速率和调制方法。

MFRC522 支持以下几种工作模式：

- 支持 ISO/IEC 14443A 的读写器模式

这些模式支持不同的传输速率和调制方法，以下章节将详细介绍这几种不同的模式。

注：本章提到的所有调制系数和调制模式都是系统参数。这表示除 IC 设置之外，要得到最优的性能还需要进行适当的天线调谐。

### 6.1 ISO/IEC1443A 功能

物理层参数描述见表 2。

表 2 ISO/IEC 14443A 读写器通信参数概述

通信方向	信号类别	传输速率		
		106kBd	212kBd	424kBd
读卡器到卡	调制	100%ASK	100%ASK	100%ASK
	位编码	修正米勒编码	修正米勒编码	修正米勒编码
	位长	128(13.56us)	64(13.56us)	32(13.56us)
卡到读卡器	调制	副载波负载调制	副载波负载调制	副载波负载调制
	副载波频率	13.56MHz/16	13.56MHz/16	13.56MHz/16
	位编码	Manchester 编码	BPSK	BPSK

## 7 MCU 接口

### 7.1 控制器接口检测

MFRC522 支持可直接相连的各种微控制器接口，如 SPI、I2C 和 UART。MFRC522 可复位其接口，并可对执行了上电或硬复位的当前微控制器接口的类型进行自动检测。MFRC522 通过复位阶段后控制管脚上的逻辑电平来识别微控制器接口。每种接口有固定管脚的连接组合。表 11 列出了不同的连接配置：

表 11 检测不同接口类型的连接方法

管脚	接口类型		
	UART (input)	SPI (output)	I <sup>2</sup> C-bus (I/O)
ALE	RX	NSS	SDA
A1	0	0	1
A0	0	1	EA
D7	TX	MISO	SCL
D6	MX	MOSI	ADR 0
D5	DTRQ	SCK	ADR 1
D4	-	-	ADR 2
D3	-	-	ADR 3
D2	-	-	ADR 4
D1	-	-	ADR 5

### 7.2 串行外设接口

MFRC522 支持 SPI 接口与主机的高速通信，接口可处理高达 10Mbit/s 的数据速率。在与主机通信时，MFRC522 作为一个从机，从外设主机上接收数据来设置寄存器，发送和接收与 RF 接口通信有关的数据。

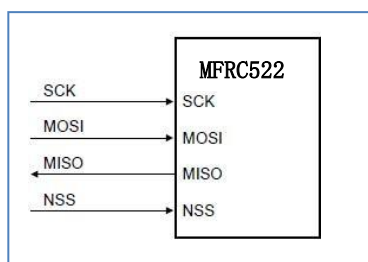


图 3 使用 SPI 接口连接到主机

在 SPI 通信中 MFRC522 作为从机。SPI 时钟信号 SCK 必须由主机产生。数据通过 MOSI 线从主机传输到从机。通过 MISO 线数据从 MFRC522 发回到主机。

MOSI 和 MISO 传输每个字节时都是高位在前。MOSI 和 MISO 上的数据在时钟的上升沿必须保持不变，在时钟的下降沿改变。

#### 7.2.1 SPI 读数据

使用表 12 所示的结构可将数据通过 SPI 接口读出。这样可以读出 n 个数据字节。

发送的第一个字节定义了模式和地址。

表 12 MOSI 和 MISO 字节顺序

管脚	字节 0	字节 1	字节 2	To	字节 n	字节 n+1
MOSI	地址 0	地址 1	地址 2	.....	地址 n	00
MISO	x <sup>[1]</sup>	数据 0	数据 1	.....	数据 n-1	数据 n

[1] X=无关项 注：先发送最高位。

## 7.2.2 SPI 写数据

使用表 13 所示的结构可将数据通过 SPI 接口写入 MFRC522。这样对应一个地址可以写入 n 个 数据字节。

发送的第一个字节定义了模式和地址。

表 13 MOSI 和 MISO 字节顺序

管脚	字节 0	字节 1	字节 2	To	字节 n	字节 n+1
MOSI	地址 0	数据 0	数据 1	.....	数据 n-1	数据 n
MISO	x <sup>[1]</sup>	x <sup>[1]</sup>	x <sup>[1]</sup>	.....	x <sup>[1]</sup>	x <sup>[1]</sup>

[1] X=无关项 注：先发送最高位。

## 7.2.3 SPI 地址

地址字节必须按下面的格式传输。

第一个字节的 MSB 位定义了使用模式。MSB 位设置为 1 时，从 MFRC522 读取数据；MSB 位设置为 0 时，将数据写入 MFRC522。第一个字节的位 6-1 定义地址，LSB 位应当设置为 0。

表 14 地址字节格式

7 (MSB)	6	5	4	3	2	1	0(LSB)
1=读	地址						0
0=写							

## 7.3 UART 接口

### 7.3.1 连接到主机

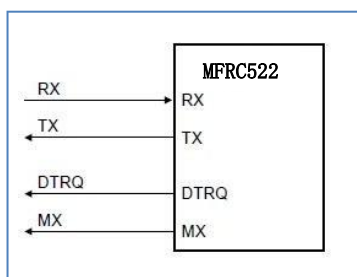


图 4 使用 UART 接口连接到微控制器

注：通过对 TestPinEnReg 寄存器的 RS232LineEn 位清零，信号 DTRQ 和 MX 可以

禁止。

### 7.3.2 UART 传输速度

内部 UART 接口兼容 RS232 串行接口。

默认的传输速率为 9.6kBaud。要改变传输速率，主机控制器必须向 SerialSpeedReg 寄存器写入一个新的传输速率值。位 BR\_T0[2:0]和位 BR\_T1[4:0]定义的因数用来设置 SerialSpeedReg 中的传输速率。

BR\_T0[2:0]和 BR\_T1[4:0]的设置参考表 15。表 16 列举了一些传输速率和相应的寄存器设置。

表 15 BR\_T0[2:0]和 BR\_T1[4:0]的设置

BR_Tn	位 0	位 1	位 2	位 3	位 4	位 5	位 6	位 7
BR_T0 参数	1	1	2	4	8	16	32	64
BR_T0 范围	1-32	33-64	33-64	33-64	33-64	33-64	33-64	33-64

表 16 可选的 UART 传输速率

传输速率(kBd)	SerialSpeedReg 值		传输速率精度 (%) <sup>[1]</sup>
	十进制	十六进制	
7.2	250	FAh	-0.25
9.6	235	EBh	0.32
14.4	218	DAh	-0.25
19.2	203	CBh	0.32
38.4	171	ABh	0.32
57.6	154	9Ah	-0.25
115.2	122	7Ah	-0.25
128	116	74h	-0.06
230.4	90	5Ah	-0.25
460.8	58	3Ah	-0.25
921.6	28	1Ch	1.45
1228.8	21	15h	0.32

[1]所有描述的传输速度中传输速度误差的结果都小于 1.5% 表 16 中所列的可选传输速率可根据下面的公式计算得到:

If BR\_T0[2:0] = 0:

$$transfer\ speed = \frac{27.12 \times 10^6}{(BR\_T0 + 1)}$$

If BR\_T0[2:0] > 0:

$$transfer\ speed = \left( \frac{27.12 \times 10^6}{\frac{(BR\_T1 + 33)}{2^{(BR\_T0 - 1)}}} \right)$$



### 7.3.3 UART 帧格式

表 17 UART 帧格式

位	长度	值
起始位	1 位	0
数据位	8 位	数据
结束位	1 位	1

注：对于数据和地址字节，LSB 位必须最先发送。传输过程中不使用奇偶校验位。读数据：

使用表 18 中的结构，可使用 UART 接口将数据读出。发送的第一个字节定义了模式和地址。

表 18 读数据字节顺序

管脚	字节 0	字节 1
RX (管脚 24)	地址	-
TX (管脚 31)	-	数据 0

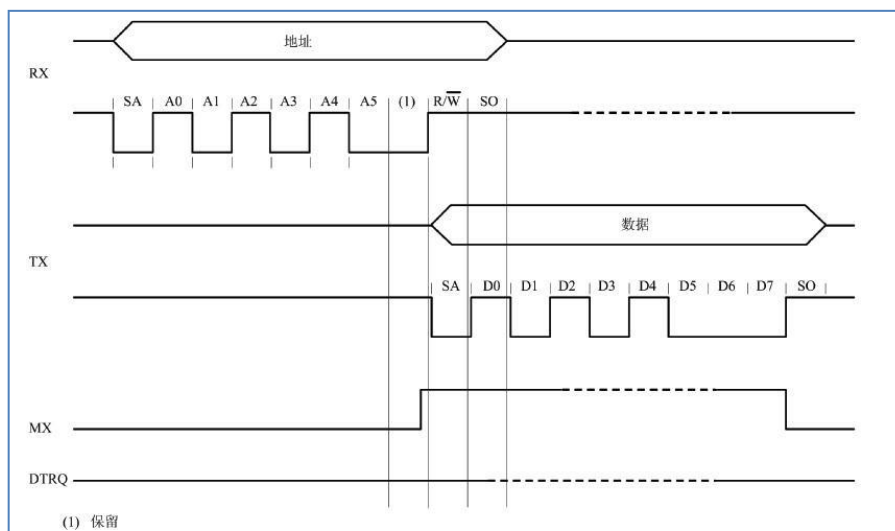


图 5 UART 读数据时序图

写数据：

使用表 19 中的结构，可使用 UART 接口将数据写入 MFRC522。发送的第一个字节定义了模式和地址。

表 19 写数据字节顺序

管脚	字节 0	字节 1
RX (管脚 24)	地址 0	数据 0
TX (管脚 31)	-	地址 0

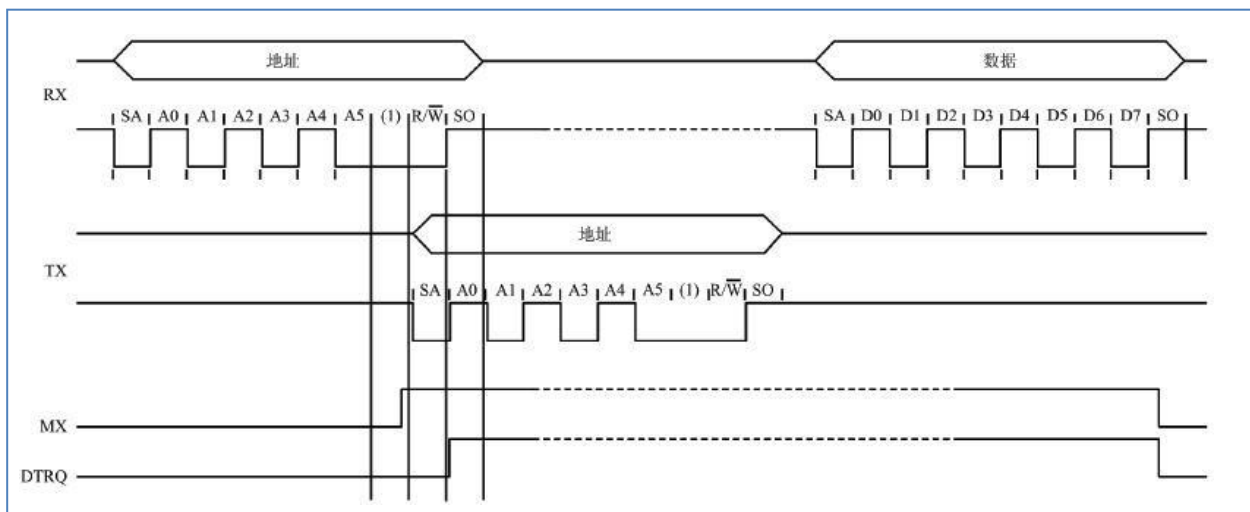


图 6 UART 写数据时序图

注：在地址字节到达 RX 管脚后，数据字节可以直接发送

地址字节：地址字节按下面的格式传输：

第一个字节的 MSB 位设置使用的模式。MSB 位设置为 1 时，从 MFRC522 读取数据。

MSB 位设置为 0 时，将数据写入 MFRC522。第一个字节的位 6 保留为将来使用，位 5-1 定义地址；详见表 20。

表 20 地址字节

7 (MSB)	6	5	4	3	2	1	0(LSB)
1=读	保留	地址					
0=写							

## 7.4 I2C 接口

支持 I2C 总线接口可以使得主机可以用较少的管脚连接到 MFRC522。I2C 接口操作遵循 I2C 总线接口规范。该接口只能工作在从机模式。

### 7.4.1 连接到主机

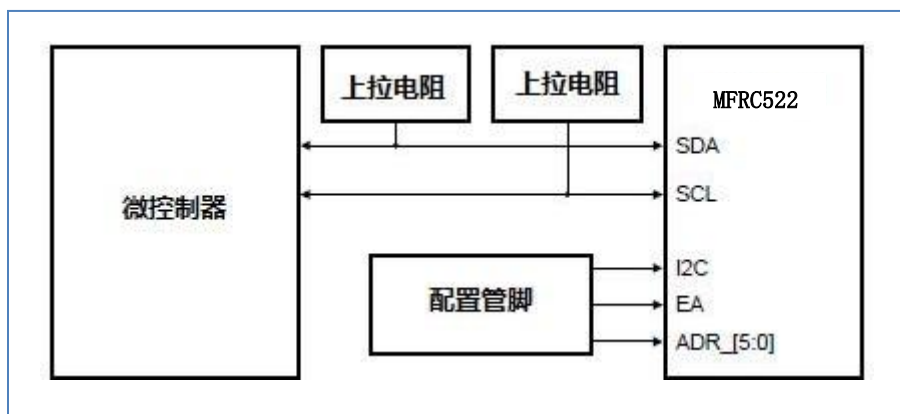


图 7 I<sup>2</sup>C 总线接口

在标准模式、快速模式和高速模式中，MFRC522 可用作从接收器或从发送器。SDA 是一个双向数据线，通过一个上拉电阻连接到正电压。不传输数据时，SDA 和 SCL 均为高电平。标准模式下，I2C 总线的传输速率为 100kBd，快速模式下为 400kBd，高速模式下为 3.4Mbit/s。

如果选择 I2C 总线接口，管脚 SCL 和 SDA 管脚具有符合 I2C 接口规范的尖峰脉冲抑制功能。

### 7.4.2 数据有效性

SDA 线上的数据在时钟周期的高电平期间保持不变。只有当 SCL 上的时钟信号为低电平时，数据线的高电平或低电平状态才能改变。

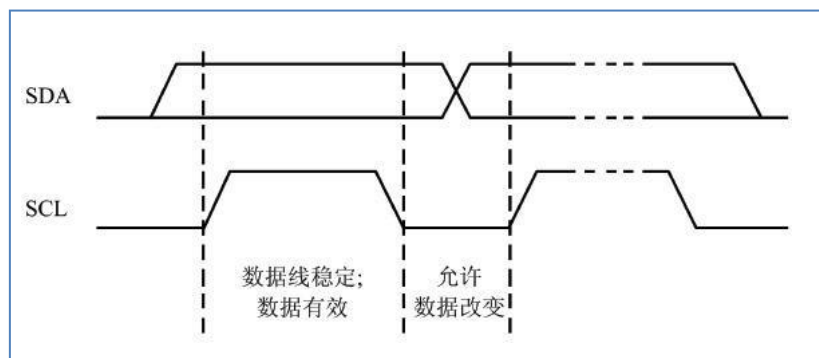


图 8 I<sup>2</sup>C 总线的位传输

### 7.4.3 起始和 停止条件

要处理 I2C 总线的数据传输，必须定义起始（S）和停止（P）条件。

- 起始条件定义为 SCL 高电平时 SDA 线上高到低的跳变。
- 停止条件定义成 SCL 高电平时 SDA 线上低到高的跳变。

起始和停止条件通常由主机产生。起始条件后主机被认为处于忙碌状态；主机在 停止条件结束一段时间后被认为重新回到空闲状态。

如果产生的是重复起始条件（Sr）而非停止条件，则总线仍处于忙碌状态。这时， 起始条件（S）和重复起始条件（Sr）的功能完全相同。因此，S 符号就用作一个常用术语，代表起始（S）和重复起始（Sr）条件。

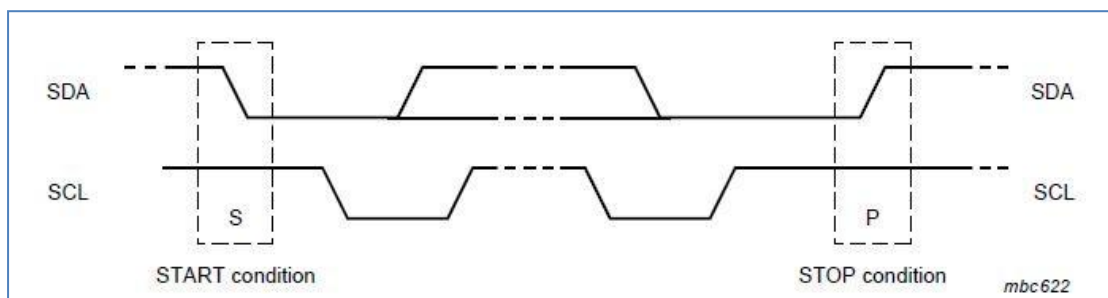


图 9 起始和停止条件

### 7.4.4 字节格式

每个字节后面必须跟一个应答位。数据传输时高位在前。一次数据传输发送的字节数不限，但必须符合读/写周期格式。

### 7.4.5 应答

应答是在一个数据字节结束后强制产生的。应答相应的时钟脉冲由主机产生。在 应答时钟脉冲周期内，数据发送器释放 SDA 线（高电平）。在应答时钟脉冲期间，接收器拉低 SDA 线使得它 在该时钟脉冲的高电平时间内保持低电平。

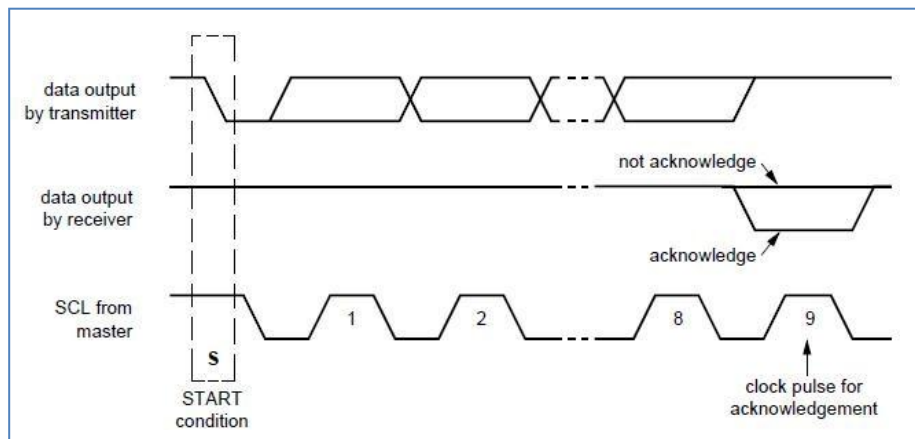


图 10 I2C 总线的应答

主机可以产生一个停止（P）条件来终止传输，也可以产生一个重复起始（Sr）条件来启动一次新的传输。

主接收器通过在最后一个字节后不产生应答来向从发送器指示数据的结束。从发送器应当释放数据线以允许主机产生一个停止（P）或重复起始（Sr）条件。

#### 7.4.6 7 位地址

在I2C 总线寻址过程中，起始条件后的第一个字节用来确定所要选择的通信从机。I2C 总线地址与 EA 管脚的定义有关。在 NRSTPD 管脚释放或上电复位后，器件根据 EA 管脚的逻辑电平来决定 I2C 总线地址。

如果 EA 管脚为低电平，则对于所有 MFRC522 器件，器件总线地址的高 4 位固定为 0101b。器件总线地址剩余的 3 位 (ADR\_0, ADR\_1, ADR\_2) 可由用户自由配置，这样就可以防止与其它 I2C 器件产生冲突。

如果 EA 管脚设置为高电平，则 ADR\_0-ADR\_5 完全由表 13 中的外部管脚来确定。ADR\_6 总是设置为 0。

在这两种模式下，外部地址编码都在复位条件释放后立即锁定。不考虑使用管脚上的进一步变化。通过配置外部连线，I2C 总线的地址管脚还可用作测试信号的输出。

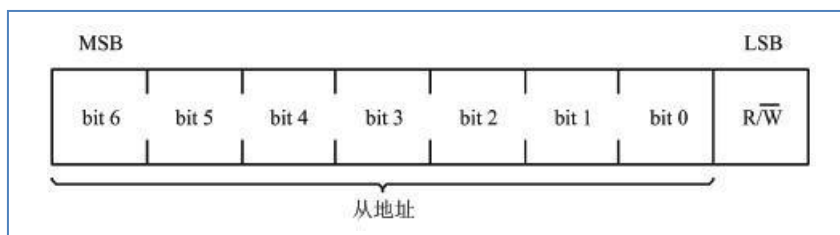


图 11 起始条件后的第一个字节

#### 7.4.7 寄存器写访问

使用下面的帧格式可用 I2C 接口将数据从主机控制器写入 MFRC522 中指定的寄存器。

- 帧的第一个字节是遵循 I2C 规则的器件地址。
- 第二个字节是寄存器地址，接下来是 n 个数据字节。

在一帧中，所有数据字节都被写入相同的寄存器地址。这种方法可使能 FIFO 的快速访问。读/写位应当设置为 0。

#### 7.4.8 寄存器读访问

使用下面的帧格式可读出 MFRC522 中指定寄存器的数据。

- 帧的第一个字节是遵循 I2C 规则的器件地址。
- 第二个字节是要读取的寄存器地址。
- 读/写位为 0。

写地址操作完成后，开始读访问。主机发送 MFRC522 的器件地址。作为回应，MFRC522 发送读访问寄存器的内容。在一帧中，所有数据字节都从相同的寄存器地址读出。这种方法可使能 FIFO 的快速访问或寄存器查询。

读/写位应当设置为 1。

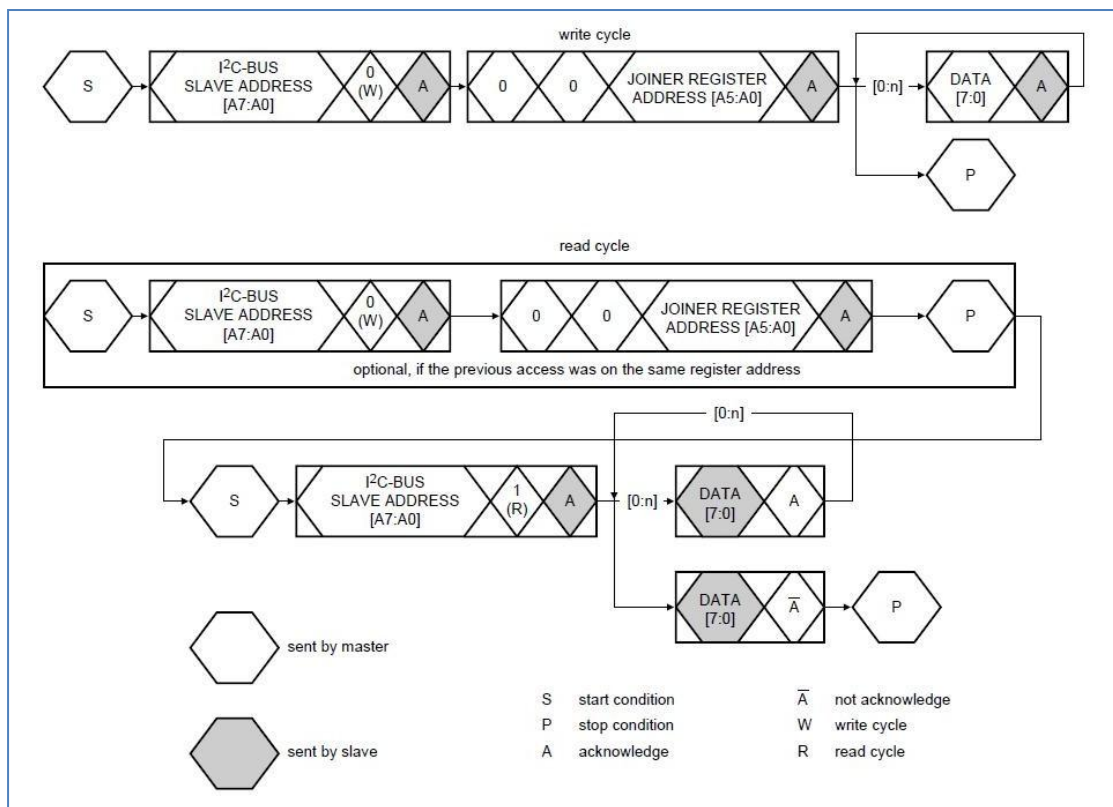


图 12 寄存器读和写访问

## 7.4.9 高速模式

在高速模式下（HS 模式），器件的传输位速率高达 3.4Mbit/s。在混合速率的总线系统中，它保持完全向下兼容快速或标准模式（F/S 模式）的双向通信。

### 7.4.10 高速传输

为了获得高达 3.4Mbit/s 的位传输速率，对 I2C 总线操作作了以下改进。

- 高速模式下器件的输入端有尖峰脉冲抑制功能，在 SDA 和 SCL 输入端有一个施密特触发器，它们与 F/S 模式相比有不同的时序常数。
- 高速模式下器件的输出端控制 SDA 和 SCL 信号的下降沿斜率，它们与 F/S 模式相比有不同的时序常数。

### 7.4.11 高速模式下的串行数据传输格式

高速模式下的串行数据传输满足标准模式的 I2C 总线规范。只有满足以下条件高速模式传输才能启动（所有条件均在 F/S 模式）：

- 起始条件（S）
- 8 位主机代码（00001XXXb）
- 非应答位（A）

高速模式开始后，主机再发送一个重复起始条件（Sr），并紧跟 7 位从机地址和一个读/写位，然后从选择的 MFRC522 接收一个应答位（A）。

下一个重复起始条件（Sr）后继续执行高速模式的数据传输，只在停止条件（P）后切换回 F/S 模式。为了降低主机代码开销，主机可以将大量的高速传输链接到一起，这些传输通过重复起始条件（Sr）分隔开来。

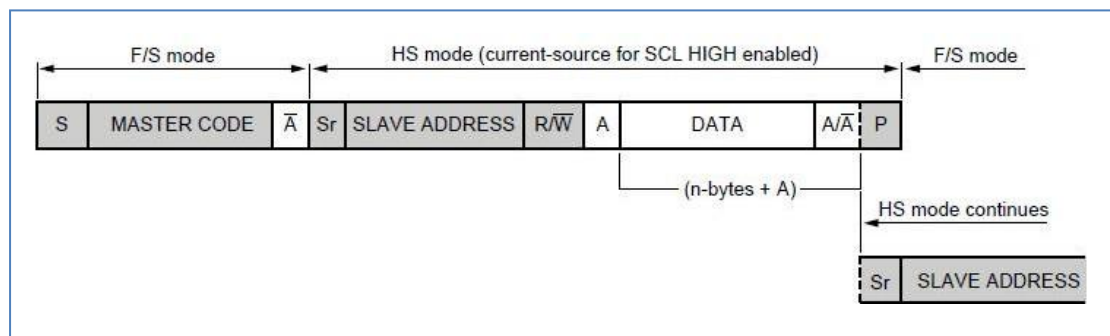


图 13 I2C 总线高速模式协议转换

#### 7.4.12 F/S 模式和 HS 模式之间的切换

复位和初始化后，MFRC522 工作在快速模式（当快速模式向下兼容标准模式时，它实际上就是 F/S 模式）。连接的 MFRC522 识别到“S 00001XXX A”序列时，会将其内部电路从快速模式设置转换成高速模式设置。

执行以下操作：

- 根据 HS 模式的尖峰脉冲抑制要求来调整 SDA 和 SCL 的输入滤波器。
- 调整 SDA 输出级的斜率控制。

对于通信中不含有其它 I2C 器件的系统配置，可以通过另一种方法永久地切换到 HS 模式。就是将 Status2Reg 寄存器的 I2CForceHS 位设置为 1。进入永久 HS 模式后，这就无需再发送主机代码了。这个操作不符合总线规范，只能用在总线无其它器件连接的情况下。此外，由于减少了尖峰抑制，一定要避免 I2C 总线上的尖峰。

#### 7.4.13 低速模式下的 MFRC522

MFRC522 完全向下兼容，可连接到 F/S 模式的 I2C 总线系统。由于此配置中不发送主机代码，因此器件处于 F/S 模式，以 F/S 模式的速率进行通信。

## 7.5 8 位并行接口

MFRC522 支持 2 中不同类型的 8 位并行接口，Intel 和 Motorola 兼容模式。

### 7.5.1 支持的主控接口

MFRC522 支持直接连接到各种微控制器。下表显示了 MFRC522 支持的并行接口类型。

表 21 支持的接口类型

支持的接口类型	总线	独立的地址和数据总线	复用的地址和数据总线
独立的读/写选通 (Intel 兼容)	控制	NRD, NWR, NCS	NRD, NWR, NCS, ALE
	地址	A0... A3[. . A5]	AD0... AD7
	数据	D0... D7	AD0... AD7
复用的读/写选通 (Motorola 兼容)	控制	R/NW, NDS, NCS	R/NW, NDS, NCS, AS
	地址	A0... A3[. . A5]	AD0... AD7
	数据	D0... D7	AD0...AD7



## 8 中断请求系统

MFRC522 通过置位寄存器 Status1Reg 的 IRq 位或激活 IRQ 管脚来指示中断。IRQ 管脚的信号可使主机使用其中断处理机制来处理。这使得软件执行效率大为提高。

### 8.1 中断源概述

表 28 列出了可使用的中断位，相应的中断源及中断产生的条件。ComIrqReg 寄存器的 TimeHRq 中断位指出一个由定时器产生的中断，当定时器从 1 减到 0 时，此中断位被置位。

ComIrqReg 寄存器的 TxIRq 位表明发送器发送完成。如果状态从发送数据变到发送结束帧，则发送器自动置位相应中断位。CRC 协处理器在处理完 FIFO 缓冲器里所有的数据后置位 DivIrqReg 寄存器的 CRCIRq 位，通过 CRCReady 位置 1 来指示。

ComIrqReg 寄存器的 RxIRq 位表明检测到接收数据的结束。如果执行完一个指令且 CommandReg 寄存器的 Command[3:0] 位的内容变为空闲时，则 ComIrqReg 寄存器的 IdleIRq 位被置位。

当 HiAlert 位置 1 且 ComIrqReg 寄存器的 HiAlertIRq 位置位时，表明 FIFO 缓冲器已经达到 WaterLevel [5:0] 位指示的长度。

当 LoAlert 位置 1 且 ComIrqReg 寄存器的 LoAlertIRq 位置位时，表明 FIFO 缓冲器已经达到 WaterLevel [5:0] 位指示的长度。

ComIrqReg 寄存器的 ErrIRq 位表示非接触式 UART 在发送或接收过程中检测到一个错误。当 ErrorReg 寄存器中的任何一个位置 1 时都表明产生了错误。

表 22 中断源

中断标志	中断源	触发动作
TimerIRq	定时器单元	定时器从 1 计到 0
TxIRq	发送器	数据发送结束
CRCIRq	CRC 协处理器	FIFO 缓冲器的数据处理完毕
RxIRq	接收器	数据接收结束
IdleIRq	ComIRQReg 寄存器	指令执行结束
HiAlertIRq	FIFO 缓冲器	HFO 缓冲器快溢出时
LoAlertIRq	FIFO 缓冲器	HFO 缓冲器快为空时
ErrIRq	非接触式 UART	检测到一个错误

## 9 定时器单元

MFRC522 有一个定时器单元，外部主机可以使用它来处理定时任务。定时器可使用如下所示的

任何一个定时/计数配置：

- 超时计数器
- 看门狗计数器
- 秒表
- 可编程一次触发
- 周期性触发器

定时器单元可用来测量两个事件之间的时间间隔或指示某段时间后指定事件的发生。它可由下文解释的事件来触发。定时器不会影响任何内部事件，例如，数据接收过程中的定时器超时并不会影响接收过程的自动处理。此外，一些与定时器相关的位可以用来产生中断。

定时器的时钟振荡频率为 13.56MHz，它是 27.12MHz 的石英晶体振荡器分频得到的。定时器包括两个阶段：预分频和计数。

预分频器(TPrescaler)是一个 12 位计数器。它的重装值(TReloadVal\_Hi [7:0] 和 TReloadVal\_Lo[7:0])在 0 到 4095 之间，由 TModeReg 寄存器的 TPrescaler\_Hi [3:0] 位和 TPrescalerReg 寄存器的 TPrescaler\_Lo [7:0]来设置。

计时器中的 16 位重装值在寄存器 TReloadReg 中定义，取值范围为 0 到 65535。

定时器的当前值在寄存器 TCouterValReg 中显示。

当计数值达到 0 时，自动产生一个中断，通过置位 CommonIRQReg 寄存器的 TimerIRQ 位来指示。如果使能，IRQ 管脚就会出现此中断信号。TimerIRQ 位可由主机来置位和复位。根据配置，定时器可以在计数到 0 时停止运行，或将 TReloadReg 寄存器的值作为初始值重新启动计数。

定时器的状态由 StatusIReg 寄存器的 TRunning 位来指示。

定时器的启动和停止可分别由 ControlReg 寄存器的 TStartNow 和 TstopNow 位来控制。

定时器还可通过设置 TModeReg 寄存器的 TAuto 位为 1 来自动激活，以满足特定的协议要求。

定时过程中的延迟时间为重装值加 1。

例如：为了得到一个 25us 的延迟，需要计数 339 个时钟周期且 TPrescaler 的值为 169。该配置使计数器每 25us 周期计数到 65535。

## 10 节电模式

### 10.1 硬掉电模式

当管脚 NRSTPD 为低电平时进入硬掉电模式。该模式下，关闭包括振荡器在内的所有内部电源。所有数字输入缓冲器和输入端分离，并关闭其功能(NRSTPD 管脚除外)，输出管脚也保持在高电平或低电平。

### 10.2 软掉电模式

CommandReg 寄存器的 PowerDown 位设为 1 后立刻进入软掉电模式。关闭包括振荡器缓冲器在内的所有内部电源。但是数字输入缓冲器不和输入端分离，且功能保持不变。数字输出管脚的状态不变。

在软掉电期间，所有的寄存器的值，FIFO 的值和配置都保持不变。

在设置 PowerDown 位为 0 后，经过 1024 个时钟周期退出软掉电模式。PowerDown 位设置为 0 并不能立刻将其清除，而是在退出软掉电模式后自动清零。

如果使用了内部振荡器，必须考虑它是由管脚 AVDD 提供的电源，必须经过一段时间( $T_{osc}$ )后，振荡器才能稳定，且内部逻辑才能检测到时钟周期。在使用串行 UART 通信时，推荐先发送 55h 给 MFRC522，振荡器必须保持稳定后才能再进一步访问寄存器。为了确保这一点，直到 MFRC522 回应上一个寄存器内容为地址 0 的读命令时才执行对地址 0 的读访问。这样说明 MFRC522 可以执行进一步的操作。

### 10.3 发送器掉电模式

发送器掉电模式切断内部天线驱动器来关闭 RF 场，可以通过设置 TXControlReg 寄存器的 TXIRFEn 或 TX2RFEn 位为 0 来实现。

### 10.4 LPCD 模式

MFRC522 内部集成低功耗自动寻卡与定时唤醒功能，寻卡时间间隔与寻卡时间均可编程，寻卡过程无需 MCU 操作，寻卡成功后可以中断唤醒 MCU。LPCD 功能在低功耗的同时，又实现了检卡，兼顾了功耗与功能。

## 11 振荡器电路

MFRC522 的时钟用作系统的编码器和解码器的时钟基准。因此，时钟频率的稳定性是保证系统良好性能的重要因素。为了获得最佳性能，必须尽可能减少时钟抖动。最好采用一个带有推荐电路的内部振荡缓冲器。

如果采用外部时钟源，时钟信号必须连接至 OSCIN 管脚。在这种情况下，特别要注意验证时钟的占空比，时钟抖动以及时钟信号的质量。

## 12 MFRC522 寄存器

### 12.1 寄存器位

根据寄存器功能的不同，寄存器的访问条件也多种多样。原则上具有相同特性的 1 位归为同组。表 23 描述了寄存器的访问条件。

表 23 寄存器位的特性

缩写	特性	描述
r/w	读/写	微处理器可以对这些位进行读和写操作，由于它们仅用于控制，其内容不会受到内部状态机的影响。例如，微处理器可以对寄存器 ComEnReg 进行读或写操作，但内部状态机对该寄存器只能读取而不能改变它们的值。
dy	动态	微处理器可以对这些位进行读或写操作，不过内部状态机同样可以改变这些寄存器的值。例如，寄存器 CommandReg 在指令执行后自动改变其内部的某些值。
r	只读	这些寄存器的值只能由内部的状态决定。例如 CRCReady 位只能表示内部的状态，外部或内部状态机都不能改变它的值。
w	只写	这些寄存器的位读结果总是 0。
RFU	-	这些寄存器是被保留的以备将来之用，在写操作时最好将这些寄存器都写为 0。
RFT	-	这些寄存器是被保留的以备将来之用或是为了生产测试。

### 12.2 寄存器

表 24 寄存器总览

地址	寄存器名称	功能
Page 0: 命令和状态寄存器组		
0h	PageReg	页选择寄存器
1h	CommandReg	掉电和命令寄存器
2h	ComEnReg	中断请求控制寄存器
3h	DivEnReg	中断请求控制寄存器
4h	ComIrqReg	中断请求位寄存器
5h	DivIrqReg	中断请求位寄存
6h	ErrorReg	指令执行的错误状态寄存器
7h	Status1Reg	通信状态寄存器
8h	Status2Reg	接收器和发送器状态寄存器
9h	FIFODataReg	64 字节 FIFO 缓冲器
Ah	FIFOLevelReg	FIFO 缓冲器已存储字节的数量寄存器
Bh	WaterLevelReg	FIFO 缓冲器溢出和空警告寄存器

Ch	ControlReg	其它项控制寄存器
Dh	BitFramingReg	面向位的帧的调节寄存器
Eh	CollReg	检查产生位冲突的第一个位的地址
Fh	RFU	保留
Page 1: 命令寄存器组		
0h	PageReg	页选择寄存器
1h	ModeReg	定义发送和接收通用模式的寄存器
2h	TxModeReg	定义发送过程的数据传输速率和结构的寄存器
3h	RxModeReg	定义接收过程中的数据传输速率和结构的寄存器
4h	TxControlReg	控制天线驱动器管脚 TX1 和 TX2 的寄存器
5h	TxAutoReg	控制天线驱动器设置的寄存器
6h	TxSelReg	选择天线驱动器的内部信号源的寄存器
7h	RxSelReg	选择内部接收器的设置的寄存器
8h	RxThresholdReg	选择位解码器的阈值的寄存器
9h	DemodReg	定义解调器的设置的寄存器
Ah	FelNFC1Reg	定义接收数据包的有效长度范围的寄存器
Bh	FelNFC2Reg	定义接收数据包的有效长度范围的寄存器
Ch	MifNFCReg	控制 ISO/IEC 14443A 和 NFC 目标模式在 106kbit 速率下的通信的寄存器
Dh	ManualRCVReg	允许手动细调内部接收器的寄存器
Fh	SerialSpeedReg	选择串行 UART 接口的速率寄存器
Page 2: 配置寄存器组		
0h	PageReg	页选择寄存器
1h	CRCResultReg	显示 CRC 计算的 MSB 和 LSB 值
2h		
3h	GsNOffReg	驱动器关闭时, 天线驱动管脚 TX1 和 TX2 上用于调制的电导寄存器
4h	ModWidthReg	控制调制宽度的设置寄存器
5h	TxBitPhaseReg	调整 106kbit 速率下 TX 位的相位寄存器
6h	RFCfgReg	配置接收器增益和 RF 电压的寄存器
7h	GsNOnReg	驱动器打开时, 天线驱动管脚 TX1 和 TX2 上用于调制的电导寄存器
8h	CWGSPReg	在无调制期间, 天线驱动管脚 TX1 和 TX2 上用于调制的电导寄存器
9h	ModGsPReg	在调制期间, 天线驱动管脚 TX1 和 TX2 上用于调制的电导寄存器
Ah	TModeReg	内部定时器的设置寄存器
Bh	TPrescalerReg	

Ch	TReloadReg	定义 16 位定时器的重载值寄存器
Dh		
Eh	TCounterValReg	16 位定时器的计数值寄存器
Fh		

Page 3: 测试寄存器组

0h	PageReg	页选择寄存器
1h	TestSel1Reg	通用测试信号配置寄存器
2h	TestSel2Reg	通用测试信号的配置和 PRBS 控制寄存器
3h	TestPinEnReg	使能 D0-D7 的输出驱动器 (注: 仅用于串行接口)
4h	TestPinValueReg	定义管脚 D0-D7 用作 I/O 总线时的值
5h	TestBusReg	内部测试总线的状态寄存器
6h	AutoTestReg	数字自检寄存器
7h	VersionReg	软件版本寄存器(读出为 0xB2)
8h	AnalogTestReg	管脚 AUX1 和 AUX2 输出寄存器
9h	TestDAC1Reg	TestDAC1 的测试值寄存器
Ah	TestDAC2Reg	TestDAC2 的测试值寄存器
Bh	TestADCReg	ADC 中 I 和 Q 通道寄存器
Ch-Fh	RFT	保留用于产品测试

## 13 指令集

### 13.1 概述

MFRC522 的运行由能够执行一系列指令的内部状态机决定的。通过把指令代码写入 CommandReg 寄存器来执行相应的指令。

### 13.2 通用特性

- 除 Transceive 指令外，需要输入数据流或（数据字节流）的指令会立即处理 FIFO 缓冲器的数据。执行 Transceive 指令时，通过设置 BitFraming 寄存器的 StartSend 位来启动传送器。

- 需要预先设置参数的指令只有当从FIFO 缓冲器中接收到正确数量的参数时才开始运行。

- 当指令启动时 FIFO 缓冲器不会立即清零， 可以先把指令参数和数据写进 FIFO 缓冲器后再启动指令。

- 新写进 CommandReg 寄存器的指令将中断当前正在执行的指令。

### 13.3 指令总览

表 25 指令总览

指令	指令代码	含义
Idle	0000	无动作；取消当前执行的指令
Config	0001	保留
Generate RandomID	0010	产生一个 10 字节的随机 ID 数据
CalcCRC	0011	启动 CRC 协处理器
Transmit	0100	发送 FIFO 缓冲器的数据
NoCmdChange	0111	不中断正在执行的指令，用来修改 CommandReg 寄存器中不影响命令执行的一些位，例如 PowerDown 位
Receive	1000	启动接收器电路
Transceive	1100	发送 FIFO 缓冲器中的数据到天线并在发送后自动启动接收器
-	1101	保留
SoftReset	1111	软复位

## 14 电气参数

### 14.1 极限参数

表 26 极限参数

参数	最小值	最大值	单位
$V_{AVDD}, V_{DVDD}, V_{PVDD}, V_{SVDD}$	-0.5	+5.5	V
$V_{TVDD}$	-0.5	+5.5	V
存储温度	-40	85	°C
ESD(HBM)	-	2000	V
ESD(MM)	-	200	V

### 14.2 主要参数指标

表 27 主要参数指标

符号	参数	条件	最小值	典型值	最大值	单位
$V_{AVDD}$	模拟电源	$V_{PVDD} \leq V_{AVDD} = V_{DVDD} \leq V_{TVDD}$	2.5	3.3	5.5	V
$V_{DVDD}$	数字电源	$V_{PVDD} \leq V_{AVDD} = V_{DVDD} \leq V_{TVDD}$	2.5	3.3	5.5	V
$V_{TVDD}$	TVDD 电源	$V_{PVDD} \leq V_{AVDD} = V_{DVDD} \leq V_{TVDD}$	2.5	3.3	5.5	V
$V_{PVDD}$	PVDD 电源	$V_{PVDD} \leq V_{AVDD} = V_{DVDD} \leq V_{TVDD}$	2.5	3.3	5.5	V
$V_{SVDD}$	SVDD 电源		2.5	-	5.5	V
$T_{amb}$	环境温度		-40	-	+85	°C
$I_{pd}$	掉电电流	$V_{AVDD} = V_{DVDD} = V_{TVDD} = V_{PVDD} = 3.3V$				
		硬掉电: NRSTPD=0	-	-	5	uA
		软掉电: RF 检测器打开	-	-	10	uA
$I_{DVDD}$	数字电源电流	$V_{DVDD} = 3.3V$	-	1	2	mA
$I_{LPCD}$	低功耗寻卡电流	400ms 自动寻卡间隔		10uA	20uA	uA
$I_{AVDD}$	模拟电源电流	$V_{AVDD} = 3.3V; RcvOff=0$	-	2	3	mA
		接收器关闭 $V_{AVDD} = 3.3V;$ $RcvOff=1$	-	1	2	mA
$I_{TVDD}$	TVDD 电源电流	管脚 TVDD;连续波	-	60	100	mA